Prompting completion

The following example shows how command-line completion works in Bash. Other command line shells may perform slightly differently.

First we type the first three letters of our command:

```
fir
```

Then we press Tab ↹ and because the only command in our system that starts with "fir" is "firefox", it will be completed to:

```
firefox
```

Then we start typing the file name:

```
firefox i
```

But this time introduction-to-command-line-completion.html is not the only file in the current directory that starts with "i". The directory also contains files introduction-to-bash.html and introduction-to-firefox.html. The system can't decide which of these filenames we wanted to type, but it does know that the file must begin with "introduction-to-", so the command will be completed to:

```
firefox introduction-to-
```

Now we type "c":

```
firefox introduction-to-c
```

After pressing Tab ↹ it will be completed to the whole filename:

```
firefox introduction-to-command-line-completion.html
```

In short we typed:

fir Tab ↹ i Tab ↹ c Tab ↹

This is just eight keystrokes, which is considerably less than 52 keystrokes we would have needed to type without using command-line completion.

Rotating completion

The following example shows how command-line completion works with rotating completion, such as Windows's CMD uses.

We follow the same procedure as for prompting completion until we have:

firefox i

We press Tab ↹ once, with the result:

firefox introduction-to-bash.html

We press Tab ↹ again, getting:

firefox introduction-to-command-line-completion.html

In short we typed:

firTab ↹iTab ↹Tab ↹